Department of Defense

Year 2000 Management Plan

Version 1.0

Office of the Assistant Secretary of Defense

19 May 97

POC: Ruby Harney
(703) 604-1475.

Office of the Assistant Secretary of Defense

(Command, Control, Communications, and Intelligence)

April 1997

Executive Summary

Introduction

This Year 2000 (Y2K) Management Plan provides the strategic guidance for all information technology, software and systems in Department of Defense (DoD) that face a "Y2K problem." The "Y2K problem" is the term used to describe the potential failure of information technology (IT) prior to, on or after January 1, 2000. This potential exists because of the widespread practice of using two digits, not four, to represent the year in computer databases, software applications, and hardware chips. Y2K related difficulties will arise when that year is "00" and our information technology will be unable to differentiate it from the year 1900. The associated but unrelated calendar anomaly that must be included in Y2K systems repairs is the fact that Y2K is a leap year unlike most other centuries.

The DoD Y2K Management Plan lays out the:

a. Goals and objectives of the DoD Y2K program
b. Overall management strategy
c. Five-Phase management process to be followed
d. Minimal performance indicators
e. Key responsibilities of the DoD Components and the DoD Chief Information Officer
f. Timeline for completion of each Y2K phase
g. DoD System Inventory and Quarterly Reporting Requirements

DoDÆs Y2K Management Strategy

The DoD has an extensive inventory of hardware and software in use today. This inventory is the result of more than three decades of IT investment. The DoD Year 2000 Management Plan focuses on Y2K resolution efforts throughout the DoD to ensure that no DoD systems failures occur before, on or after January 1, 2000, due to Y2K related problems. It provides the overall DoD strategy and guidance for inventorying systems, prioritizing systems, retiring systems, and monitoring progress.

The DoDÆs overall goal is to provide a DoD-wide coordinated effort that ensures no system is adversely affected by Y2K problems. This will be accomplished through a DoD management strategy that calls for centralized policy and decentralized implementation using five phases, Awareness,

Assessment, Renovation, Validation and Implementation, supported by program and project management activities. This goal will allow the DoD Components the flexibility to implement solutions as deemed appropriate while benefiting from best practices in a coordinated effort.

Roles and Responsibilities

The Assistant Secretary of Defense for Command, Control, Communications and Intelligence, as the Chief Information Officer (CIO) of the Department of Defense will oversee the DoDÆs solution to the Y2K problem. DoD Components are responsible for assessments, renovations, validations and implementation actions. To assist in the Y2K cross-functional issue resolution process, the Deputy Secretary for Defense has established the Y2K Steering Committee and the DoD Y2K Working Group.

Guidance for DoD Components

A Year 2000 Cost Factors Checklist and a Year 2000 Compliance Checklist are provided as guidance for the DoD Components.

Foreword

The Year 2000 (Y2K) date processing problem in the Department of Defense (DoD) poses an enormous management and technical challenge. The problem stems from the use of two-digit year fields instead of four digit year fields in software, hardware, and firmware. The effect of this will be that many computer programs will fail as they attempt to calculate against the year "00" not recognizing that the year is actually 2000. The resulting inaccuracies in date-related calculations will generate corrupt data results and potentially cause computer systems to fail entirely. Also, if erroneous information goes unrecognized, the problem is perpetuated through interfaces with other automated information systems. While this is the crux of the problem, it is more complex; many systems have faulty date logic that does not recognize that the Y2K is a leap year, other systems have triggers that are executed based on specific values of date fields, and others have overflow or rollover problems. Finally, Y2K problems are happening today and could increase in occurrence as we approach the Y2K.

The Y2K problem is not restricted to any one functional area within the DoD. The DoD uses computers to support the performance or actually perform every function it conducts. This includes business functions such as financial management, personnel management, health care, contract management, logistics management, and many other business functions. Computers also perform or support the performance of our strategic and tactical operations such as mobilizing the force, deploying the force, and maneuvering the force. Computers are used to support intelligence, surveillance, and security efforts. Also, the DoD relies heavily on computers to support weapons and weapons systems deployment. In many instances, there is a computer embedded in or integral to a weapon. When the computer fails, the weapon or weapon system fails. Many of our Y2K problems in weapon systems are

being fixed during normal maintenance. It is our automated information systems that have legacy databases and rely heavily on dates and date-related calculations that will consume the majority of our resources and efforts to fix. In view of the magnitude and seriousness of the problem, the DoD must assure that its systems supporting operational missions and decision-making functions continue to perform as designed. This document provides the DoD strategy and management approach to satisfactorily address the Year 2000 date processing problem within the DoD. The Plan outlines responsibilities and milestones and provides guidelines for Year 2000 activities to ensure that no DoD system fails due to Year 2000 problems. Because of the dynamic nature of the Year 2K problem, this Plan will be updated, as necessary.

Emmett Paige, Jr.

Table of Contents

Appendix B: Year 2000 Compliance Checklist

Appendix C: March 12, 1997 Memo and Quarterly Reporting Requirements Spreadsheets

Appendix D: Acronyms

Appendix E: Glossary

Appendix F: References

I. Introduction

1.1 Year 2000 (Y2K) is the term used to describe the potential failure of information technology (IT) systems prior to, on or after January 1, 2000. This potential exists because of the widespread practice of using two digits, not four, to represent the year in computer databases, software applications, hardware, and microchips. Difficulties will arise in the Y2K when our systems will be unable to differentiate it from the year 1900. The associated, but unrelated, calendar anomaly that must be included in the Y2K systems repairs is the fact that Y2K is a leap year unlike most other century dates.

1.2 The Assistant Secretary of Defense for Command, Control, Communications and Intelligence, as the Chief Information Officer (CIO) of the Department of Defense, has the responsibility to lead DoD efforts to solve the Y2K problem. DoD Component Heads are responsible for making sure that all software correctly process dates. The ASD(C3I) and other senior leaders in the DoD will resolve the DoDÆsY2K problem through execution of a five phase process*: Awareness, Assessment, Renovation, Validation, and Implementation.*

1.3 Most experts agree that Y2K resolution efforts begin with a thorough assessment (inventory) of existing systems. The DoD has an extensive inventory of hardware and software in use today. This inventory consists of more than three decades of IT development. The goal is to have all DoD systems certified ascertified as Y2K compliant and implemented not later than November 1, 1999. This will be accomplished through the elimination, replacement and/or modification of existing systems as they move through the phases. The new FAR 39.106 states that compliance is: "information technology that accurately processes date/time data (including but not limited to, calculating, comparing, and sequencing) from, into, and between the twentieth and twenty-first centuries, and the years 1999 and 2000 and leap year calculations. Furthermore, Year 2000 compliant information technology, shall accurately process date/time data if other information technology properly exchanges date/time data with it."

II. Purpose and Scope

2.1 Purpose

The DoD Year 2000 Management Plan focuses on Y2K resolution efforts throughout the DoD. It provides the overall DoD strategy and guidance for inventorying systems, prioritizing systems, retiring systems, and monitoring progress. The DoDÆs CIO has overall responsibility for overseeing the DoDÆs solution to the Y2K problem. DoD Components are responsible for awareness, assessments, renovations, validations, and implementation actions. The DoDÆs overall goal is to provide a DoD-wide coordinated effort that ensures no system is adversely affected by Y2K problems prior to, on, and after January 1, 2000.

2.2 Scope

This Management Plan applies to the Office of the Secretary of Defense (OSD), the Military Departments, the Chairman of the Joint Chiefs of Staff, the Combatant Commands, the Inspector General of the Department of Defense (IG, DoD), the Defense Agencies, and the DoD Field Activities (hereafter referred to collectively as "the DoD Components"). It shall apply to all interfaces between the DoD and external organizations including other Government agencies, the private sector, non-profit organizations, allies, coalition partners, NATO, and other alliances.

This Plan applies to the Defense Information Infrastructure (DII), all systems supported by information technology, their technical environment, and their communications devices, including but not limited to automated business information systems, automated command and control systems, and weapon systems. Information technology support includes hardware, firmware, commercial off the shelf (COTS), Government off the shelf (GOTS) developed software, and data. Software includes COTS/GOTS packages, operating systems, third and fourth generation language compilers and interpreters, functional applications, system utilities, translators, and database management systems (DBMSs). Data includes databases, files, and other data storage structures and mechanisms, data and system interfaces and interchanges, Electronic Data Interchange (EDI) transaction sets and implementation conventions, and other messages or forms of data exchange.

III. Goal and Objectives

1. Year 2000 Goal and Objectives

The DoD goal is to ensure no system failures occur due to Y2K related problems. Objectives include:

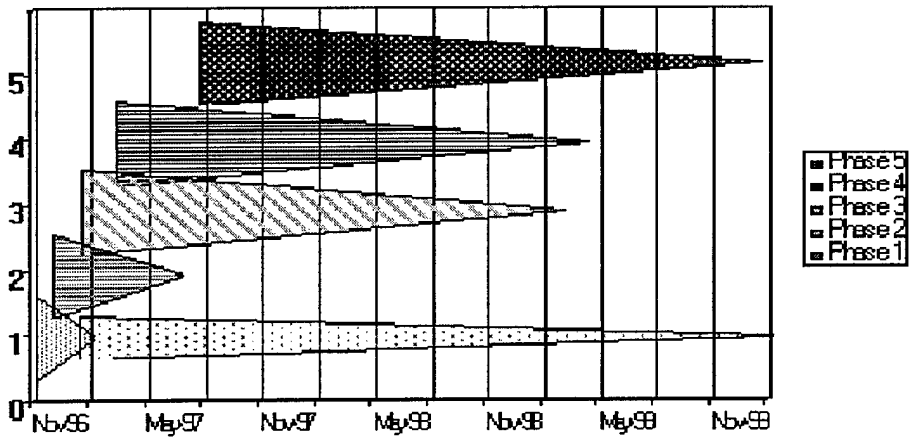
- Minimize the adverse impact of Y2K date processing in all mission and mission support systems.
- Define and share DoD-wide, consistent strategies for finding and fixing Y2K problems, and testing solutions.

- Minimize duplication of effort for finding and fixing Y2K problems, and testing solutions.
- Minimize the impact of resource reallocation to support Y2K efforts.
- Minimize risk and cost in determining the appropriate Y2K solution for each system.
- Recognize the Y2K problem as an opportunity to retire legacy systems early.
- Identify, prioritize, and mobilize needed resources for system conversions and replacements.

IV. Management Strategy

4.1 The DoD management strategy calls for centralized policy and DoD wide implementation using the five phase process, supported by program and project management activities. This allows Components the flexibility to address the problem as they see fit while also benefiting from best practices in a well-coordinated effort. Y2K efforts are underway and require parallel execution of portions of our five-phase solution process. The five phases are:

1. Awareness. The first phase focuses on promoting Y2K awareness throughout DoD.
2. Assessment. The second phase consists of system inventory and problem assessment.
3. Renovation. The third phase constitutes systems replacement, retirement, or conducting repairs to ensure Y2K compliance.
4. Validation. The fourth phase in which systems are tested for Y2K compliance and interoperability.
5. Implementation. The fifth and final phase is systems deployment.



5 Phase Timeline

-

-

4.2 Decide on an Overall Approach

The DoD ComponentÆs overall approach is being disseminated throughout the organization. It is essential that DoD Components establish an approach that addresses each phase of the Y2K resolution process. As an example, the DoDÆs management plan focuses on centralized management with decentralized execution. The DoDÆs CIO is responsible for overseeing the Y2K solutions. DoD Components are responsible for the awareness, assessments, renovations, validations, and implementation actions. The DoD Components are also responsible for reallocation of resources to fix their systems impacted by Y2K problems. Oversight and DoD wide implementation is dependent upon the following:

3. Information Sharing

To reduce duplication of effort and leverage Y2K experiences, information on Y2K problems, best practices, and lessons learned will be shared. The primary sharing media for this effort are the Y2K public and restricted homepages on the Internet. This sharing of information will include sharing with other DoD Components, Government agencies, and the private sector.

4. Completion Target

The DoD target for completing of all Y2K efforts is November 1, 1999. However, we would it is expected that most systems will be compliant well before this date.

5. Resourcing

The management strategy is that existing resources will be used for Y2K compliance efforts. The DoD recognizes Y2K efforts may cause delays of some change request proposals or preplanned product improvements. Fixing Y2K problems is the DoDÆs top software resource priority, nonessential software sustainment requirements, enhancements, preplanned product improvements, and change request proposals will be immediately postponed until all systems have been analyzed, fixed, tested, and verified Y2K compliant using the attached checklist (Appendix B). Systems funded with post deployment software support will use these moneys to fund the Y2K effort.

6. Prioritization

Systems that are critical to the support of DoD warfighting and peacekeeping missions (e.g., weapons systems, command and control) and those that affect the safety of individuals shall receive priority for conversion and replacement. Migration systems should receive priority for assessment and correction of identified Y2K problems based on criticality of the system.

7. DoD Standardystem Date Format

DoD Components should use a 4-digit contiguous year for the year portion of dates used for interfaces among systems and in all interagency information exchanges. The four digit date format is required for systems interfaces and data exchanges in DOD to reduce the risk of reinfection of Y2K problems in the DepartmentÆs systems and databases. In Electronic Commerce (EC)/Electronic Data Interchange (EDI) transactions, where other formats are used, the Components will use four digit year representations when they are available, will use the century indicator ("CC") when it is available, and will use translators as necessary. The century indicator is the first two digits of a four digit year (CCYY). Those systems using an ordinal date format must ensure the Y2K compliance. If a system is Y2K compliant but does not use a standard date format, conversion to the standard is not required at this time.

8. DoD System Termination/Retirement/Elimination

Legacy systems or systems that do not support the organizational mission or ones that could be combined into other systems, are prime candidates for termination. Additionally, this is an opportunity to eliminate unnecessary systems from the inventory. System termination is the preferable solution to a Y2K problem if a risk assessment determines that it is a viable option. In some cases, the "window of vulnerability" (period during which dates will be improperly processed) of a system is small. It may be decided that the system will not be used during that period, or that a temporary "workaround" solution can be applied during the "window" and removed afterward.

9. Replacement Alternatives

DoD will take advantage of Y2K compliant Commercial off-the-shelf (COTS) or Government off-the-shelf (GOTS) solutions whenever practical to replace a system that has Y2K problems. Another replacement alternative is to rapidly redevelop the system through rapid application development (RAD), rapid architected application development (RAAD), Business Process Reengineering (BPR), or object technologies and methodologies.

V. The Five Phase Management Process

5.0 The five phase management approach was developed by the Air Force and has since been adopted by GAO and other Government Agencies.

5.1. AWARENESS PHASE

The DoD is aware of the Y2K problem. Its potential impact on the DoD mission requires continuous and ongoing awareness.

5.1.1 Define the Problem

Defining the problem is relatively straightforward and is most likely accomplished. The "Y2K problem" is the term used to describe the potential failure of information technology (IT) prior to, on or after January 1, 2000. This potential exists because of the widespread practice of using two digits, not four, to represent the year in computer databases, software applications, and hardware chips. Difficulties will arise in the Y2K when that year is 00 and our information technology will be unable to differentiate it from the year 1900. It affects everyone, and its deadline is fixed!

5.1.2 Establish a Component Y2K Office

The first step in attacking Y2K is to establish a Component level Y2K point of contact (POC). The Component head must decide the level of resources and attention needed for Y2K solutions. The first objective should be to develop and publicize a Y2K corporate approach outlining the ComponentÆs high level strategy. This approach will serve as the basis for the ComponentÆs detailed action plan.

Once accomplished, efforts should shift to focusing on the Awareness and Assessment Phases. While the first phase may be incomplete, it will still provide much needed guidance and direction throughout the organization. Attempting to prioritize and schedule systems, procure resources, and allocate funding requires management decision at the Component level. While the actual "fixes" will take place in the trenches, placing those who will make these fixes in position to do so requires significant planning at the highest level.

5.1.3 Identify Technical and Management Representatives

Another area critical to successful Y2K resolution is the identification of technical and management points of contact (POCs). This includes system managers, budgeting and resource personnel, legal representatives, senior management, support contractors and other external contacts.

5.1.4 Desktop and Distributed Computing Systems

DoD has in recent years seen considerable increase in the utilization of desktop computing and distributed computing environments. The interfacing and data exchange between such various distributed computing systems must be addressed for Y2K problems to ensure proper data handling and conversion for the Y2K. Components need to determine dependency links between internal and external systems, between core mission areas, processes and all data exchange entities, and provide for date and data format conversions where necessary. Data bridges and filters may provide some of the solutions for external links, but a validation process is necessary to ensure compliance. The checklist in Appendix B may assist in this process.

5. DoD Contracts

DoD contracts should follow the guidance provided in the Federal Acquisition Regulations, 48 CFR Parts 39.002 that addresses Y2K compliance definitions and language. All contracts will conform to the following principles:

1. The DoD will purchase only Y2K compliant products.

2. The DoD will use standardY2K compliance language in contracts.

3. The DoD will issue stop work orders on all contracts for new products being purchased on existing contracts for products that fail to meet Y2K requirements.

4. Contracting offices will request contractors develop a Y2K compliance plan to upgrade their Y2K non-compliant products.

5.1.6 Y2K Compliance Certification

System developers/maintainers along with the systemÆs functional proponent will certify and document each systemÆs Y2K compliance. A sample Y2K compliance checklist is at Appendix B. Signature by the System Manager on the checklist constitutes æcertificationÆ of Y2K compliance for each system.

5.2 ASSESSMENT PHASE

This phase deals with those activities required to define the scope of the problem and set up the infrastructure necessary to solve it. The primary deliverable from this phase is the Project Plan for specific systems within the Components.

The primary purpose of the Assessment Phase is to gather and analyze the information in order to determine the size and scope of the problem. Only after the size and scope of the problem has been determined can an estimate of the cost in terms of dollars and work years be made.

DoD Components are to continually assess the impact of Year 2000 on their Information Technology hardware, software, and devices to include the Year 2000 impact caused by microchips. The results are to be reported quarterly in accordance with the "Year 2000 (Y2K) Refined Reporting Requirements for DoD" and the spreadsheet "Quarterly Reporting Requirements for Y2K Assessment and Progress-DoD." Both are in Appendix C.

5.2.1 Code Inventory

The Code Inventory involves locating all of the code that must be modified for the Year 2000. The volume and type of code will determine the magnitude of the problem. Source code may be housed in a single repository, or the ownership and responsibility for maintenance may be decentralized and spread over the work force. Having the code in a centralized repository of some type, whether itÆs an elaborate vendor library package or a simple partitioned data set, is a big help in solving the Y2K problem.

All code must be inventoried and tracked and its relationship to other code determined. A total count of the lines of code (LOC) will assist in determining how many and what type of resources will be required in order to make the changes. Organizing code by application will assist in the preparation of the detail Component plan. This will help in the identification of applications which exchange data so that changes to these applications can be scheduled for the same time.

It is imperative to understand what language code is written in.

For example, if you decide to buy a Y2K tool to help with this effort, the tool must work with the language(s) you have. Ownership of code is important in order to determine who is responsible for making the Y2K changes and certifying that each piece of code is Y2K compliant.

In order to modify code for the Year 2000, you must have programmers skilled in the language the code is written. Knowing what percentage of your code is written in each language aids you in determining if you have the correct skill mix.

5.2.2 Collecting Survey Information

An assessment survey can be used to gather the necessary information about source code which is not contained in a central repository. The compliance Checklist (Appendix B) may be used to aid system managers in ensuring their systems are Y2K compliant. In addition, it will aid in the identification of COTS products that may be imbedded within other products. The survey process seeks to collect system data in a standard format. Once the information is collected, it can be stored in a central database or spreadsheet. Collecting data such as system IDs, descriptions, components (language, platform, etc.), interfaces, owners/users/maintainers, and other relevant information is critical to any future efforts such as prioritizing and scheduling systems for renovation. Every system, even those currently selected for migration or retirement, must be inventoried.

5.2.2.1 The checklist assessment survey should be distributed to all Components. It will yield several important results:

5.2.2.1.1 The checklist survey provides the first indication of the level of effort that must be accomplished. When the results of the checklists survey are summarized, a very rough cost estimate can be applied using $1.10 per executable line of code (ELOC).a metric developed by the Gartner Group and MITRE. For embedded weapon systems, the Components should use $8.00 per ELOC. These metrics allow the DoD Components to estimate costs for the entire life cycle, including evaluating the system, doing the repairs, and fully testing the system. As assessments progress, more detailed estimates based on projected engineering costs, person-hours, and testing requirements should be used.

5.2.2.1.2 The checklist survey may also identify that common components exist which can be managed from the central project team. The collection and distribution of information regarding in-house and vendor tools and services is a good example. Knowing which platforms and languages are most common will allow Y2K management to focus their efforts more precisely, such as in the evaluation of tools for analyzing systems written in specific languages.

5.2.2.1.3 Finally, even if a central repository is in existence for the bulk of ComponentÆs code, it is not uncommon, due to the proliferation of personal computers and local area networks, for individual departments to develop their own computer systems. The survey will help to identify these systems so that awareness, assessment, and renovation can be extended to them. Additionally a survey which is properly worded will generate input from those infrastructure systems (for example, environmental such as heating, security, and elevators) which must also be date tested.

5.2.3 Missing Source Code

Missing source code increases both the scope and cost of the project because it requires time and resources to develop both the functional specifications and program specifications in order to rewrite the missing modules. Many Components will discover, as they inventory their code, that there are some programs that are running for which the source code is no longer available. In some cases these programs have been running for years. Unfortunately the Y2K issue requires that every piece of code be examined to determine if any two digit date handling is involved. This will may require the re-creation of all programs that have no source code.

Three ways to address this issue include:There are several ways to address this issue. First, assess the impact of failure of this system and determine if there is already a replacement system in development. If there is no replacement, but the impact of failure is low, then the cost of rewriting or disassembling the object code is not cost effective. Instead, plan for how to deal with the possible failure. However, if there is no replacement, but the impact of failure is high, then consider one of the following options:

1. Assign the task to in-house programmers to have them either rewrite the code from the original specifications (if available) or disassemble the object code and try to re-create the source from that.

2. Send the object code to a vendor who has the ability to re-create source through a combination of existing software tools and proprietary products. In both cases the final result should be source code that can be examined for Y2K compliance and is easily maintainable.

3. Replace existing system or termination of the system all together.

5.2.4 Mapping Source To Executables

A one-to-one mapping of source code to executable code must should be made in order to determine that the source code in the inventory actually corresponds to the executable code running in production. There are several products on the market that allow you to map your source code to your executables. In general, they each produce a report that breaks all the load modules in a particular load library and displays pertinent information about the source code.

5.2.5 Vendor Software

All the operating system software and program products that surround the application software may need changing. The first step in this process is compiling a comprehensive list of vendor software used by your component. Currently there is no comprehensive listing of Y2K compliant software. MITRE maintains a listing of software but the software has not been verified for compliance. The Component must determine if the vendor software is Y2K compliant.

5.2.6 Contractor Maintained Software

Some Components will also have to deal with the issue of contractor maintained software. Software in this category will need to be modified for the Y2K and may be done by the contractors.

5.2.7 Pilots

As the source code issues are resolved and an inventory is established, consideration of the budget and scheduling issues is required. System owners, users, designers, and developers cannot assume any system is Y2K compliant until it has been certified certified. Estimating the amount of time required to actually make a module/program Y2K compliant is not exact. It will depend upon the complexity of the program, whether documentation exists, the skill level of the programmer, and the familiarity of the programmer with the program in question. One process is to conduct a pilot, maintaining careful records of the time required to modify the code for Y2K compliancy. The code selected should be representative of the bulk of the inventory. The time spent can then be extrapolated over the total inventory, to produce a work year estimate. If more than one pilot is conducted, using modules of varying complexity, the work year estimate gained can then be weighted across the inventory, depending upon the percentage of code by complexity.

Another reason for conducting a pilot is to determine if a Y2K tool would be beneficial. Piloting an application without the aid of a tool and comparing the results against the same or similar code modified with the assistance of one or more of the tools available will show whether savings can be achieved. Another reason for conducting a pilot would be to determine which approach should be used when changing the code. Several applications could be piloted using different approaches and the results compared to see which is the most cost effective.

Be aware, the use of pilots can consume a considerable amount of time and resources.

5.2.8 Identify Technical Issues

At this point all other technical issues that could affect the project should be identified. Consider them as

you develop your cost estimates, risk management plans, and contingency plans. Examples of these issues follow and will affect the cost of solving the Y2K problem:

- Forms -- pre-printed and computer generated
- On-site vs. off-site contractor resources (off-site adds costs)
- Screen issues (2 or 4 position years)
- Library clean-up
- Format of dates on inputs and outputs
- Standardized date routines
- Databases and archives

5.2.9 Estimating System Costs

There are a number of factors that will influence the cost of making systems Y2K compliant in addition to modifying software. They include building the test environment, buying tools and services, adding hardware, upgrading operating system software and commercial products, etc. The DoD has developed a checklist for "estimating system costs for the Y2K," which includes those additional items that must be considered. The checklist will indicate those areas where costs should be adjusted because of your specific environment. See Year 2000 Cost Factors Checklist at Appendix A.

5.2.10 Estimating System Costs For Y2K

-

If the component has developed a more accurate means of developing a cost estimate, they may use it.

The DoD is using a combination of cost metrics developed by the Gartner Group and MITRE Corporation and internal Component estimates. For embedded weapon systems, the cost algorithm is $8.00 per executable line of code (ELOC). For all other systems, $1.10 per ELOC is applied. These metrics allow the DoD to estimate costs for system evaluation through validation and implementation.

As stated earlier, as assessments progress, systems managers must provide more detailed estimates based on projected engineering costs, person-hours and testing requirements should be used as they become available. Also DoD Components may base cost estimates on in-house cost models or actual fixes. The Components must identify the methodology used.

5.2.11 Tools

Testing tools, re-engineering tools, and other types of tools may be necessary at various times. There are two main types of tools that are being marketed to deal specifically with the Y2K problem. The first type consists of those tools that change the system date for an individual batch job. This allows Components

to determine how specific programs or systems will handle processing for any chosen future date.

Tools that help locate dates and related fields and in some cases change those dates in the source code comprise the second category. These tools usually provide a pre-defined set of fields such as "century", "year", "yy", etc. to which fields known to be dates in specific systems can be added. The tools then examine the code for occurrences of these fields and also track the flow of dates as they move from field to field. The output produced by these tools helps pinpoint those areas in the code that must be examined most closely.

There is no silver bullet, i.e., a tool that will find all date fields in the code and make the necessary changes. No one has developed such a product and most experienced programmers do not believe one is possible given the nearly infinite possibilities for storing and manipulating dates.

The LIST OF YEAR 2000 VENDORS and TOOLS on the Defense Information Systems Agency (DISA) homepage will provide more information on those vendors who have tools that fall into the two categories described here (see Appendix FD, Federal Y2K Web sites).

5.2.12 Procurements

Procurements may be required at any point during the Y2K project. An overall procurement strategy should be developed during this phase and refined in later phases. Tools to be used during the renovation phase should be bought here. This includes tools that help with the analysis, tools that allow system date manipulation, and common date routines. In addition, once the assessment has been completed, it will be possible to determine whether additional manpower will be needed, or whether the project can be completed with existing resources.

5.2.13 Risk Analysis/Prioritization

Only after an inventory of source code has been completed and a rough estimate of the size and scope of the problem within a Component has been obtained, can decisions regarding priorities be made. When the data gathering portion of the assessment phase has been completed and resource estimates are available, some Components will realize that it is impossible to do everything.

When work cannot be completed, Components will need to undertake a risk analysis of their systems. Each system will need to be assigned a priority based on its mission criticality. Those Components that get a late start on the problem and/or do not have the necessary infrastructure in place may find themselves with not enough time to convert and test even critical systems.

5.2.14 Develop Validation Approach

The Assessment Phase should include the development of a strategy and a validation schedule. The schedule should indicate the general time frames for the validation of all systems and should consider hardware concerns such as availability of processing cycles and storage, along with human resource issues.

Future date testing should be done in isolation with no possibility of corrupting or destroying production files. If the resources need to be available in late 1997 or 1998, the appropriate reprogramming and procurement actions need to start. Validation should be completed as soon as possible.

5.2.15 Electronic Data Interchanges

Electronic Data Interchanges (Interfaces) involve the sending and receiving of data between Services and/or Defense Agencies or external DoD vendors, etc. The National Institute of Standards and Technology issued a FIPS (Federal Information Processing Standard) Publication Change notice on March 25, 1996, which stated: "for purposes of electronic data interchange in any recorded form among U.S. Government agencies", NIST highly recommends that four-digit year elements be used. The year should encompass a two-digit century that precedes, and is contiguous with, a two-digit year-of-century. In addition, optional two-digit year time elements specified in ANSI x3.30-1985 (R1991) "should not be used for the purposes of any data interchange among U.S. Government agencies."

Electronic Data Interchanges are critical in the Y2K effort because they have the potential to introduce and/or propagate errors from one DoD Component to another. Where this cannot be coordinated in a timely fashion, bridges can be written to accommodate the transition period. Bridges receive information in one format, modify it, and put it out in another format, such as receiving the year in a 2 digit format, adding century information through the use of an algorithm, and writing the output with a 4 digit year. Where the bridge must accommodate data from a number of sources, it can be table driven, to accept the input in either format, based upon an entry in the table, but output the data in a fixed 4 digit year format.

When bridges are written as separate modules/steps, they can be more easily removed when the sending/receiving programs become fully compliant. Existing documentation should be updated to reflect the insertion of bridges or the use of filters, sliding windows and algorithms.

5.2.16 Developing a Plan

The ComponentÆs plan, started during the Awareness Phase, should be completed as the last step in the Assessment Phase. The plan should show, at a minimum, the start date and release date for each phase, the major steps to be taken in converting and testing the code and establishing the necessary infrastructure, and the resources required to accomplish these tasks. Beyond these basics, the project plan can be developed to whatever degree of detail is necessary at each Component.

The most challenging aspect of the plan will probably be the scheduling of each system. This will require tremendous coordination within the Component and between Components that exchange data. Where possible, the goal should be to convert related systems simultaneously to reduce the number of bridge programs that must be written and maintained.

5.2.17 Develop contingency plans

Unlike routine system development or maintenance efforts where schedule slippages are non-fatal--and common--the Y2K program must be completed on time. Components should develop realistic contingency plans, including the development and activation of manual or contract procedures, to ensure the continuity of their core processes. Contingency plans should be updated at each phase.

5.3 RENOVATION

The Renovation Phase involves making and documenting software and hardware changes, developing replacement systems, and eliminated systems. Renovation involves conversion of an existing application; replacement deals with the development of a new application; elimination focuses on the retirement or decommissioning of an existing application or system component. In all three cases, the process must also consider the complex interdependencies among applications, hardware platforms, databases, and the internal and external interfaces.

Organizations are required to identify the strategy they are using for renovation. If they are not using field expansion, they are then required to identify -- and report in the quarterly reporting -- the reason behind their strategy. Whenever possible, Components should have this information posted on the DISA Y2K homepage.

All changes to the information systems and their components must be made using configuration management procedures to ensure that changes are adequately documented and coordinated throughout the agency. Equally important is the need for each agency to assess dependencies and to communicate all changes to the information systems to internal and external users.

5.3.1 The processes that may be in the Renovation Phase are:

-

1. Conversion

   Convert selected applications, databases, archives, and related system components. In converting application systems, consider changes in operating systems, compilers, utilities,

domain-specific program products, and commercial database management systems.

1. Develop data bridges and filters

Ensure that all internal and external data sources meet the Y2K date standards of the converted or replaced systems. Develop bridges to convert non-conforming data. Filters may be used to edit out nonconformant data.

1. Replace

Replacement deals with the development of a new application or expansion of an existing application. Selected applications, platforms, database management systems, operating systems, compilers, utilities, and other COTS software which are not in compliance may be replaced. Ensure that replacement products are Y2K compliant, including their ability to properly handle the leap year adjustments. Contract specialists and legal staff are to review contracts and warranties.

1. Document code and system changes

Implement and use configuration management procedures to ensure that all changes to information systems and their components are properly documented and managed.

1. Unit, integration, and system tests

Schedules for unit, integration, and system tests following the conversion of individual application and software modules should be completed. Coordinate scheduling with other Components to ensure that all system components, including data bridges or filters, are available for testing. Systems will likely fluctuate between renovation and validation in somewhat of a spiral or incremental cycle. Test schedules must include time for regression testing.

1. Eliminate

Selected applications, platforms, database management systems, operating systems, utilities, and COTS software may be eliminated. Prepare to terminate applications, platforms, database

management systems, operating systems, utilities, and COTS software not being replaced.

1. Communicate changes

It is necessary that all changes to the DoD information systems and components, the DIST, and specifically all changes to date formats for data exchanged with other systems or external organizations be communicated to DoD through the DIST, quarterly reporting and/or the appropriate DoD web pages. Document changes through the configuration management process. Communication of those changes to the user community is necessary.

1. Track the conversion and replacement

Track the conversion and replacement projects and collect and use project metrics to manage cost and schedule. This tracking should be reflected in the quarterly reports.

1. Share information

Ensure that project staffs understand the need to collect and disseminate information on lessons learned and best practices. Develop dissemination strategy and tools, such as web sites, newsletters, etc.

5.4 VALIDATION

DoD Components will need an extensive period of time to adequately validate and test converted or replaced systems for Y2K compliance, and Gartner Group estimates the testing and validation process could consume over half of the Y2K program resources. The length of the validation and test phase and its cost is driven by the complexity inherent in the Y2K problem. The DoD Components must not only test Y2K compliance of individual applications, but also the complex interactions between scores of converted or replaced computer platforms, operating systems, utilities, applications, databases, and interfaces. In some instances, DoD Components may not be able to shut down their production systems for testing, and may have to operate parallel systems implemented on a Y2K test facility.

All converted or replaced system components must be thoroughly validated and tested to (1) uncover errors introduced during the Renovation Phase, (2) validate Y2K compliance, and (3) verify operational readiness. The testing should account for application compliance, database interdependencies, and interfaces. The testing should take place in a realistic test environment. Each DoD Component should assess their testing procedures, facilities, and tools to ensure that all converted system components meet

quality standards and are Y2K compliant.

1. Develop and document tests, plans, and schedules

For each converted or replaced application or system component, Components should develop and document test and compliance plans and schedules. Components should eEstablish a compliance validation process.

2. Develop a strategy for managing the testing of contractor-converted systems

In many instances, a Component will contract for the conversion of selected systems and their system components. The contract conversion must be closely managed to ensure that the contractor follows the agency's Y2K conversion standards. In addition, the Component must ensure that the contractor-converted systems are adequately tested and certified via the compliance checklist.

3. Implement a Y2K test facility

The actual requirement of a Y2K test facility must be determined by individual Components. Where such facilities are implemented, testing the converted or replaced systems and their system components for Y2K compliance will likely require an isolated test facility capable of simulating Y2K requirements. The test facility should provide for large test databases and multiple versions of the application software.

4. Implement automated test tools and test scripts

Components may take advantage of tThe use of computer-aided software testing tools and test scripts have the potential to significantly reduce the testing and validation burden. Test management tools may help in the preparation and management of test data, in the automation of the comparison of test results, in scheduling and incident tracking, and in managing test documentation.

5. Perform unit, integration, and system testing Using a phased approach, perform unit, integration, and system testin

Use selected testing techniques to ensure that the converted or replaced systems and accompanying components are functionally correct and Y2K compliant.The testing should include regression, performance, stress, and forward and backward time testing.

5.4.6 Components should dDefine, collect, and use test metrics to manage the testing and validation process.

7. 5.4.7 Initiate acceptance testing

Acceptance testing is the final stage of the multiphase testing and validation process. During this phase, the entire information system -- including data interfaces -- is tested with operational data. Testing should be done at a Y2K test facility or in a separate environment with duplicate databases to avoid risk to the production systems and the potential contamination of data.

8. Complete acceptance testing

In general, formal testing uncovers about 80-90 percent of software errors, with the remaining 10-20 percent of errors discovered during operations. Acceptance testing should be completed no later than June 30, 1999, to allow sufficient time for the correction of software errors discovered following implementation.

-

5.4.96 Develop Update contingency plans

Unlike routine system development or maintenance efforts where schedule slippages are common but not-fatal the Y2K program must be completed on time. Components should develop realisticupdate contingency plans, including the development and activation of manual or contract procedures, to ensure the continuity of their core processes.

5.5 IMPLEMENTATION

-

Implementation of Y2K compliant systems and their components requires extensive integration and acceptance testing to ensure that all converted or replaced system components perform adequately in a heterogeneous operating environment. Because of the scope and complexity of the Y2K conversion changes, integration, acceptance, and implementation will likely be lengthy and costly.

Once converted and subsequently tested, Y2K compliant applications and system components must be implemented. Since not all system components will be converted or replaced simultaneously, Components may be expected to operate in a heterogeneous computing environment composed of a mix

of Y2K compliant and noncompliant applications and system components. The reintegration of the Y2K compliant applications and components into the ComponentÆs environment must be carefully coordinated to account for system interdependencies. Parallel processing--where the old and the converted systems are run concurrently--may be needed to reduce risk.

1. Define transition environment and procedures

The transition from the current environment to Y2K compliant systems will be difficult and complex. First, some key components of the DoD ComponentsÆ systems -- Y2K compliant databases, operating systems, utilities, and other COTS products -- may not be available until late 1998 or early 1999. Second, external data suppliers may not plan to complete their conversion and testing until 1999. Third, the testing, validation, and correction processes may take much of 1998 and may extend into 1999. Fourth, replacement systems may not be ready for testing until late 1999. As a result, Components may be forced to operate, at least for a time, parallel systems and databases.

2. Develop implementation schedule

The Y2K implementation schedule must deal with uncertainties common to all large system development efforts, and should indicate all major milestones and the critical path for the completion of the Y2K program.

5.5.3 Resolve data exchange issues and interagency concerns, including ensuring that:

- All outside data exchange entities are notified.
- Data bridges and filters are ready to handle non-conforming data.
- Contingency plans and procedures are in place if data is not received from an external source.
- Contingency plans, developed during the Assessment Phase and procedures are updated and in place if invalid data is received from an external source.
- The validation process is in place for incoming external data.

All data issues and interagency concerns must be resolved prior to acceptance testing and to implementation. Acceptance testing should be completed prior to this Phase. . Bridges and filters should be in place to handle non-conforming data received from external sources, and contingency plans and procedures should be in place to handle no data or bad data situations.

4. Deal with database and archive conversion

Because the conversion of large databases from 2 - digit to 4 - digit year fields is a time consuming effort, Components may consider off-site conversion alternatives.

5.5.5 Complete acceptance testing - In general, formal testing uncovers about 80-90 percent of software errors, with the remaining 10-20 percent of errors discovered during operations. Acceptance testing should be completed no later than June 30, 1999, to allow sufficient time for the correction of software errors discovered following implementation.5.5.57 Update or develop disaster recovery plans

All Y2K compliant systems, including the converted and replaced systems and related databases, should have disaster recovery plans for the restoration of operations and data in case of extended outage, sabotage, or natural disaster.

5.5.8 Implement converted and/or replaced systems - Reintegrate the converted and/or replacement systems and related databases into the production environmen

-

5.5.79 Post implementation considerations

Implementation completes the phases delineated in the Y2K Management Plan. However, a period of close monitoring of systems should follow through the Y2K to ensure that compliant systems do in fact work. When January 2000 arrives, any problem that surfaces may be attributed to the Y2K problem. DoD Components should be aware of the necessity of monitoring after the change in the calendar year. The possible expenditure of additional funds to correct any unforeseen problems that may occur should be part of a contingency plan. All verified problems related to the Y2K should be documented. Additionally, other key dates throughout the year should be watched carefully for Y2K related problems.

VI. Performance Indicators

1. The Y2K computing challenge comes with the perfect ultimate performance measure -

January 1, 2000. Does your system work prior to, on or on or after that date? In the meantime the DoD will measure progress towards a Y2K solution using a variety of indicators. Current performance indicators for Y2K problem solution are (but are not limited to):

- The number of systems in each phase of the five phase management process.
- The number of systems scheduled for elimination (not being replaced).

- Total estimated costs.
- Total amounts obligated.

6.2 Effective July 18, 1997, the DoD Y2K performance indicators will, in addition to the indicators above, include:

- The total number of systems in DoDÆs inventory;
- The number of systems eliminated;
- The number of systems impacted by the Year 2000 problem;
- The total number of interfaces;
- The number of interfaces impacted by the Year 2000 problem;
- The number of interfaces fixed;
- The number of systems fixed, and;
- Remaining systems to be fixed.

6.3 The Components shall submit a quarterly report to the ASD(C3I) along with a short narrative evaluation on the above indicators no later than the third week of each quarter beginning April 1997 (Appendix C).

VII. Responsibilities

The ASD(C3I), as the DoD Chief Information Officer, has the overall management responsibility for Y2K. To assist in the Y2K cross-functional issue resolution process, the Deputy Secretary for Defense established the Y2K Steering Committee and the DoD Y2K Working Group.

7.1 The Y2K Steering Committee

The Y2K Steering Committee, established December 16, 1996, oversees progress, provides guidance, and makes decisions related to the Year 2000. The Committee serves as a forum to facilitate the sharing of information, eliminate overlaps, and identify cross-functional issues or opportunities that accelerate Year 2000 system fixes. The Committee is chaired by the Deputy Secretary of Defense (DepSecDef) with the DoD CIO as the Executive Secretary. Membership on the Committee includes representatives from all the major DoD Components.

7.2 DoD Year 2000 Working Group

The OASD(C3I) DoD Year 2000 Focal Point chairs the DoD Y2K Workgroup. The Workgroup supports the activities and deliberations of the Year 2000 Committee. Each DoD Component shall assign a representative to the Workgroup to investigate Y2K and, cross-functional issues, provide recommendations, identify and share lessons learned, and avoid duplication of effort within the DoD.

7.3 ASD(C3I)

The ASD(C3I) in the role of the DoD CIO shall:

- Establish DoD-wide strategies and policy guidance for addressing the Y2K problem.
- Set Department-wide goals for completion of each phrase of the DoD Year 2000 activities.
- Oversee DoD-wide Y2K planning and implementation of Year 2000 activities across DoD and monitor progress.
- Represent the DoD in Y2K discussions with DoD and other government Components.
- Serve as the Executive Secretary to the DoD Year 2000 Steering Committee.
- Establish Y2K reporting requirements

7.4 DoD Component Heads

The Component heads or their designated Y2K point(s) of contact are responsible for the implementation of this plan and shall do the following:

- Prepare and execute a Y2K oversight program for systems under their functional area(s).
- Identify and prioritize critical systems consistent with approved management strategy.
- Discontinue or replace old applications systems, if they are determined to be too costly to repair.
- Monitor the execution of Y2K corrections for systems within their functional areas.

  For non-compliant systems, make decisions to execute Y2K corrections, replace systems, retire systems, or postpone modifications consistent with the ComponentÆs management strategy. In each case, conduct a risk assessment to ensure the associated level of risk is acceptable. Include this risk assessment in their Y2K oversight program.
- Document and obtain system interface agreements in the form of Memorandums of Agreement (MOAs) or equivalent.
- Make resource decisions and develop strategies for systems with Y2K problems within their functional area(s). Identify budget shortfalls and include them in budget submissions and reprogramming actions.
- Purchase and develop only Y2K compliant systems.
- Include Y2K compliance language in all new contracts and contract modifications as appropriate.
- Identify a Y2K POC acting as the single POC for all Y2K questions and actions within each functional area(s). POC will participate in the Y2K Work Group as required.
- Develop individual Year 2000 Plan, with milestones geared to programmatic requirements.
- In the third week of each quarter, beginning with the second quarter 1997, submit to the ASD(C3I), a short narrative evaluation of your organizationÆs Y2K efforts, responding to ASD(C3I) guidance

in Appendix C. Include in the evaluation an overall appraisal of the situation, Y2K impacts, major concerns, Y2K performance indicators, and recommendations.

7.5 Defense Information Services Agency (DISA)

- Maintain the DoD official system inventory repository using the Defense Integration Support Tools (DIST) database.
- Support the DoD CIO in executing DoD-wide Year 2000 initiatives.
- Provide technical assistance to the Components.
- Maintain a current and accurate listing of tools available to the Components which can assist in resolving the Y2K problems.

7.6 DASD(I&S) Intelligence Systems SecretariatSecretartiat (ISS) has the overall lead in coordinating efforts for the DoD Intelligence Community in addressing the Year 2000 challenge.

VIII. Timeline

8.1 The timeline is for each of the five phases. The end dates represent completion target dates for each phase. Some phases must overlap in order to complete all actions no later than November 1, 1999. The timeline presented here is consistent with the Office of Management and BudgetÆs timeline. However, it is expected that DoD Components will complete each phase as quickly and as thoroughly as possible.

This aggressive schedule is necessary due to the limiting factor in the Y2K project û Time. Components should target these dates for completion of exit criteria and beginning and completing phases.

Identify systems that cannot meet these completion targets and closely track them at the appropriate level to ensure exit criteria are completed by the last day of each phase. Additionally, ensure that viable contingency plans are in place.

8.2 Phase I (Awareness) - Awareness, education, and initial organization and planning take place.

- **Target Completion Date: Dec 1996**

-- Exit Criteria:

--- Phase I plan completed and distributed.

--- Corporate strategies developed and ready to be submitted to OSD.

--- Y2K POCs identified and educated for all organizations.

--- System users and owners identified and educated.

--- Key DoD and industry POCs contacted.

--- Phase II strategy developed, documented, and distributed.

8.3 Phase II (Assessment) - Scope of Y2K impact is identified and system level analyses take place.

- **Target Completion Date: June 1997**

-- Exit Criteria:

--- Phase II plan completed and distributed.

--- 100% inventory of all systems input into DIST (Target: Nov. 30, 1996).

--- Phase III strategy developed, documented, and distributed.

--- 100% of systems to be replaced, redeveloped, retired, identified and confirmed (Target Completion Date: March 31, 1997).

--- 100% of systems analyzed for Y2K compliance.

--- Y2K resource strategy and plan developed and completed.

--- 100% of systems requiring renovation prioritized and scheduled for Phase III.

--- Phase IV strategy developed, documented, and distributed.

--- Risk management and contingency strategy developed, documented, and distributed.

8.4 Phase III (Renovation) - Required system "fixes" are accomplished

- **Target Completion Date: Dec 1998**

-- Exit Criteria:

--- Phase III plan completed and distributed.

--- Successful implementation of selected renovation strategy for all scheduled systems.

--- Phase IV plan completed and distributed.

--- Phase V strategy developed and completed.

--- Risk management and contingency strategy updated.

8.5 Phase IV (Validation) - Systems are confirmed as Y2K compliant through assorted testing and certification compliance processes.

- **Target Completion Date: Jan 1999**

-- Exit Criteria:

--- Unit, integration, and system testing completed, systems certified.

--- Acceptance testing and certification completed.

-

8.6 Phase V (Implementation) - Systems are fully operational after being certified in Phase IV.

- **Target Completion Date: Nov. 1, 1999**

-- Exit Criteria:

--- Risk management and contingency strategy updated and distributed.

--- Systems successfully integrated and operational (Target Completion Date: Nov. 1, 1999).

IX. DoD System Inventory and Quarterly Reporting Requirements

9.0 The Defense Integration Support Tools (DIST) is the official repository for DoDÆs inventory of systems for the DoD Components. It contains information on hardware platforms, operating systems, applications languages, communications, and interfaces. DoD Components are required to report quarterly, systems to the DIST database. Although the minimum reporting requirement is quarterly, DoD Components are encouraged to report significant progress when it occurs. The reason for this reporting is to give the DoD the visibility necessary to ensure a thorough and successful transition to Y2K compliance for all DoD systems. The reporting will also keep other Functionals, that your systems interface with or exchange data with, informed as to the status of your Y2K compliance progress. Additionally, although there will be no budgetary relief to accomplish this mission, Congress has requested and will continue to pursue an aggressive total accounting of the cost of compliance. Your initial reporting must be completed by April 18, 1997. Subsequent reports are similarly due in July, October, and January 1998.

9.1 Appendix C contains a copy of the March 12, 1997, memorandum: "Year 2000 (Y2K) Refined Reporting Requirements for DoD" with the "Quarterly Reporting Requirements for Y2k Assessment and Progress" spreadsheets. Future modifications to the Plan may include modification to these requirements.

9.2 Definitions

a. Y2K System: An automated process that uses information technology such as computer hardware and software to perform a specific function, application, or service.

b. Mission Critical System: a system that when its capabilities are degraded, the

organization realizes a resulting loss of a core capability.

c. Migration System: An existing automated information system (AIS) or application, or a planned and approved AIS or application, that has been officially designated as the single AIS or application to support standard processes for a function.

d. Legacy System: An existing automated information system (AIS) or application which will be replaced in whole or part by a migration system.

3. Systems to be reported in the DIST

All systems which meet or exceed the following criteria must be reported in the DIST for Y2K purposes:

a. a mission critical system;

b. a migration system;

c. a legacy system;

d. a system with a $2M total cost per year (i.e., Tab G reported in the Information Technology Systems Budget); or,

e. a system that interfaces with a system that meets any one of the above criteria.

Data elements required for Y2K reporting apply to these systems, IAW USD(Comptroller)/ ASD(C3I) memorandum of Nov. 5, 1996, subject: System Interfaces, Data Exchanges, and Defense Integration Support Tools.

4. Non-DIST criteria systems

In addition to those systems reported in paragraph 2, above, the DoD Components will report to OASD(C3I) all other Program Executive Officer (PEO), separate Program Manager (PM), and/or major command/activity systems not meeting the above DIST criteria as a "one-line entry" with the following data:

a. number of systems compliant (do not include retired systems),
b. number of systems noncompliant (do not include retired systems),
c. number of systems being retired,
d. number of noncompliant systems, in each of the five phases, and
e. total estimated cost to fix.

5. Devices

Furthermore, the DoD Components will report to OASD(C3I) on devices controlled by information technology. Three categories of devices control information technology: PCs and servers; communications hardware/software (routers, bridges, switches, PBXs, etc.); and, facilities and other systems (biomedical equipment, HVAC, sprinklers, FAX machines, elevators, security, etc.). Aggregate data for each of these areas to be reported are:

a. number compliant (do not include retired/discontinued devices),
b. number noncompliant (do not include retired/discontinued devices),
c. estimated cost to fix (due to Y2K compliancy, not planned modernization).

APPENDIX A: YEAR 2000 COST FACTORS CHECKLIST

NOTE: Year 2000 "compliancy" includes proper processing of Leap Years [The Year 2000 is a Leap Year.] The purpose of this checklist is to aid in estimating system costs for the Y2K during the assessment phase. Go through the checklist and mark all factors that apply to your system. Next, develop relative determinations of how these factors affect your cost estimate. Finally, apply these relative determinations to refine your cost estimate. Additionally, you can use the factors checked to consider in developing your test, contingency, and risk plans.

A.1 Application Software:

____ Size: Number of executable lines of code (LOC)

____ Age: Older code tends to be less structured and thus harder to understand

____ Complexity: Relative intricateness/understandability of the business rules

____ Documentation: Degree of documentation available and its understandability

____ Programmer: Familiarity with the program code. Level of skill/competency/expertise

____ Source Code: Availability

____ Date- "Intensiveness": Relative number of date related

calculations/comparisons

____ Embedded Dates: Frequency of date use as part of data element or in data

element codes

____ Date Formats Used: Consistency within the system of a standard date format

____ Year 2000 Strategy (Field expansion/procedural code/sliding window): Different strategies to achieve Year 2000 "compliancy" have different costs.

____ Language: Some languages (e.g., COBOL 68) are unable to properly process the Year 2000 so the software will have to be upgraded/changed. [Additionally, the language relates to the availability of the Year 2000 COTS tools, programmers to work on the system, and availability of Year 2000 compliant COTS].

A.2 Hardware/System Software: Year 2000 compliancy of each of the components of the technical environment is required. [Often only a current version of a product will be Year 2000 compliant.]

____ Operating System

____ Major Subsystems: Sometimes subsystems have different technical environment components.

____ Database Management System (DBMS)

____ Compilers/cross-assemblers (available - sometimes they don't exist)

____ Teleprocessing (TP) monitors

____ Homegrown/locally developed software that is used in conjunction with the system

____ Workstation Software: Consider the quantity needed.

____ Workstation BIOS (handles the "system clock function"): 60-80% of PC BIOSs are not Year 2000 compliant -- most are soldered to the "motherboard, " some are reprogrammable, some are "socketed" and some can be replaced.

____ Programmer: Familiarity with the hardware and operating system; level of skill/competency/expertise

____ Programmer System Software (utilities and development tools): To support making changes to the software

____ Capacity/Usage Level: Making a system Year 2000 compliant may increase storage requirements or even CPU requirements and cause a need to purchase a larger computer.

____ Embedded Software (microchips/circuit cards; e.g., PABXs, security system (access control), cash registers): They may be directly or indirectly related to a system, and may not be Year 2000 compliant. The availability of compliant hardware or the cost of developing, and the quantity required must be considered.

____ Communications: Telecommunications hardware and software upon which the system depends must be considered.

____ Network Timestamps (LANIWAN network, clock time): Upon which the system is dependent

A.3 Database/Files:

____ Number of date-related data elements

____ Amount available

A.4 Year 2000 Tool Support:

____ Availability: Many languages and/or technical environments do not have Year 2000 COTS tools so tools must be developed in-house or specifically contracted for development.

____ Quality

A.5 External Interfaces/Middleware:

____ Data Sources: Must be evaluated and "bridges" planned as required.

____ Data Outputs: Must be evaluated and "bridges" planned as required.

____ DI Transaction Sets: System may generate some EDI transactions or get input from EDI transactions which require "bridges."

____ Reports: Systems may generate paper reports which need to be modified.

____ Screens: Systems may have screens used by users which require modification.

A.6 System Plans:

____ Planned Major Upgrade: May be used to do Year 2000 compliance work at the same time to reduce costs.

____ Termination: System may be eliminated before a Year 2000 problem occurs.

____ Replacement: System is planned for COTS replacement or reengineering before impacted by the Year 2000.

A.7 Miscellaneous System-Related Information:

____ Sort Routine Year 2000 compliancy

____ Backup Routine Year 2000 compliancy

____ Archival Routine Year 2000 compliancy

____ System Criticality/Priority: Really not required for cost estimate, but a good time to record this critical planning information.

____ Risk Analysis (if system fails): Really not required for cost estimate, but a good time to record this critical planning information. Consequences of system failure must be considered.

____ Risk Analysis (if system is not made Year 2000 compliant): Many systems only have a small "window of vulnerability" during which not being able to process Year 2000 properly occurs. Consideration must be given to whether or not this "window" is acceptable; i.e., the system won't be used during that period, or a "workaround" will be established for that period; e.g., manual processing.

____ Contingency and Continuity of Operations Planning

A.8 Year 2000 Management:

____ Project Management

____ Configuration Management

____ Change Management

____ Contract(or) Management

____ Year 2000 Emergency Reaction Team

A.9 Year 2000 Testing:

____ Establishing Test Environment

____ Unit Testing

____ Integrated Testing

____ Year 2000 Simulation Testing: Can sometimes require mirror of production environment. Might not be possible until technical environment is made Year 2000 compliant.

APPENDIX B: YEAR 2000 COMPLIANCE CHECKLIST

The purpose of this checklist is to aid system managers in ensuring that their systems are compliant for the Year 2000. Make sure the following items are included in your Year 2000 testing and certification compliance process for all of the developed, gratis, licensed, and purchased software, hardware, and firmware used in your systemÆs operation, development/maintenance, support, and testing activities.

Y2K compliant system accurately processes date/time date from, into and between the twentieth and twenty-first centuries and the leap year calculations. Finally, "compliant" systems have no extended semantics, calendar errors, date overflow, and inconsistent semantics.

Please respond to each question with the appropriate answer.

System Identification

(An asterisk indicates an optional question)

B.1. Please provide system information.

a.	Name of system	
b.	Defense Integration Support Tools (DIST) Number of system	
c.	Operational date of system (current or a future date)*	
d.	Planned or actual replacement date of system (retirement or discontinuation qualifies as replacement)*	
e.	For planned replacements what is the contingency plan and under what conditions will it be invoked?*	
f.	What are the safety critical portions of the system, if any?*	

Year 2000

B.2. Each system has its own window of time, before and after the present date, in which it functions. Planning and scheduling systems work with dates that are weeks, months, and sometimes years in the future. Likewise, trend analysis systems and billing systems regularly reference dates in the past. For your system, and its window of time, please verify its ability to successfully process data containing dates with no adverse effect on the applicationÆs functionality and with no impact on the customer or end user beyond adjustment to approved changes in procedures and data formats.

		VERIFIED		NO		N/A
a.	Dates in 20th century (1900s)					
b.	Dates in 21st century (2000s)					
c.	Dates across century boundary (mix 1900s and 2000s)					
d.	Crosses 1999 to 2000 successfully					

Other/Indirect Date Usage

B.3. Have you verified performance (and corrected if necessary):

		VERIFIED		NO		N/A
a.	Dates embedded as parts of other fields					
b.	Dates used as part of a sort key					
c.	Usage of values in date fields for special purposes that are not dates (e.g. using 9999 or 99 to mean "never expire")					
d.	Date dependent activation/deactivation of: passwords, accounts, commercial licenses					
e.	Date representation in the operating systemÆs file system (creation dates and modification dates of files and directories)					
f.	Date dependent audit information					
g.	Date dependencies in encryption/decryption algorithms					
h.	Date dependent random number generators					
i.	Date dependencies in firmware					
j.	Personal Computer BIOS and RTC does not reset the year to 1980 or 1984 on reboots after 31 December 1999 *(corrections by operating system utilities allowed)*					

Leap Year

B.4. System accurately recognizes and processes Year 2000 as a leap year.

		VERIFIED		NO		N/A
a.	February 29, 2000 is recognized as a valid date					
b.	Julian date 00060 is recognized as February 29, 2000					
c.	Julian date 00366 is recognized as December 31, 2000					
d.	Arithmetic operations recognize Year 2000 has 366 days					

Usage of Dates Internally

B.5. Internal application usage of dates and date fields must be clear and unambiguous in the context of the systems which use them.

		VERIFIED	NO	N/A
a.	Display of dates is clear and unambiguous (the ability to correctly determine to which century a date belongs either by explicit display, i.e. 4-digit year, or system or user inference)			
b.	Printing of dates is clear and unambiguous			
c.	Input of dates is clear and unambiguous			
d.	Input of logically correct dates			
d.	Storage of dates is clear and unambiguous			

External System Interfaces

B.6. External interactions are identified and validated to correctly function for all dates.

		VERIFIED		NO		N/A
a.	Interaction between this system and any other external time source, if existing, has been verified for correct operation.					
	For example, the GPS system is sometimes used as a time source. Many GPS receivers cannot correctly deal with the roll-over of the GPS 10-bit epoch counter that will occur at midnight, 21 August 1999. GPS receivers also deal with an 8-bit Almanac Week counter which has a 256 week roll-over span.					
b.	You and the responsible organization for each interface have negotiated an agreement dealing with Year 2000 issues.					
	For example, is the interface currently Y2K compliant, is it being worked on, does it have an unknown fix date, or will it be fixed by a future date you have mutually agreed on.					
	For each interface that exchanges date data, you and the responsible organizations have discussed and verified that you have implemented consistent Year 2000 corrections that will correctly work for date data passed between your systems.					

Date Field Type

B.7. Describe the type of date fields used by the system, in either software or data bases.

		VERIFIED	NO	N/A
a.	Does the system use 4 digit year data fields?			
b.	Does the system use 2 digit year data fields?			
c.	If 2 digit, does the system use a century logic technique to correctly infer the century?			
d.	At what date will the century logic fix fail?			
		YES		NO
e.	Are there any internal data types for dates?			

If yes to e, what is the range of dates that the date field can represent?

Minimum Date		Maximum Date	

Year 2000 Testing Information

B.8. Optional: Please provide the following information for all year 2000 compliance tests that are conducted, i.e. system test, integration test, acceptance test:

		Narrative Answer
a.	Testing Organization	
b.	Name of Test Team Chief	
c.	Date that Year 2000 compliance testing was completed	
d.	How was Year 2000 compliance determined? (certified by vendor or contractor, tested in-house, inspected but not tested, etc.)	

		YES		NO
e.	Are the test data sets available for regression testing on the next version release for questions 2, 3, 4, 5, 6, 7d, and 7e?			
f.	Are the detailed test results and reports available for review and audit for questions 2, 3, 4, 5, 6, 7d, and 7e?			
g.	Do you follow a defined process for tracking the status of all Year 2000 problems reported, changes made, testing, compliance, and return to production?			

COTS/GOTS Components

B.9. Optional: Please provide the following information with regard to COTS/GOTS components.

		YES		NO		N/A
a.	Does the system use COTS/GOTS application packages and/or infrastructure components?					
b.	If yes, have those items been verified to be Year 2000 compliant?					

		Narrative Answer
c.	How was Year 2000 compliance determined? (certified by vendor or contractor, tested in-house, etc.)	

Certification Levels

B.10 Certification levels are defined below. Yes, verified and N/A are considered positive responses. No is considered a negative response.

LEVEL

0 System retired or replaced

1 Full independent testing completed with either:

- All questions have positive responses except possibly 7b
or

- All questions have positive responses except possibly 7a

2 Independent audit of system and existing testing completed with either:

- All questions have positive responses except possibly 7b
or

- All questions have positive responses except possibly 7a

3 Self-certification

CAUTION: Self-certification assumes a higher risk level of potential failures

3a Self-certification with full use of 4 digit century date fields

- All questions have positive responses except possibly 7b

3b Self-certification indicates risk due to use of 2 digit century fields

- All questions have positive responses except possibly 7a

3c Self-certification indicates risk due to ambiguous usage of dates

- Question 5-a,b,c or d have negative responses.

3d Self-certification indicates potential problems (System needs additional work before Year 2000 processing can be assured with any level of reliability)

- Question 2-a,b,c or d have negative responses, or

- Question 3-a,b,c,d,e,f,g,h,i or j have negative responses, or

- Question 4-a,b,c or d have negative responses, or

- Question 5-a,b,c or d have negative responses, or

- Question 6-a or b have negative responses, or

- Question 9-b has a negative response.

4 Not certified or not certified yet.

B.11 It would be advisable but not required for the system/program/project manager to have the responsible programmer(s) fill out a similar checklist covering the software they are responsible for before completing this checklist for the overall application.

LEVEL OF CERTIFICATION FOR THIS DATA SYSTEM: (*Circle only one)*

0 1 2 3a 3b 3c 3d 4

I certify that the information provided above is true and correct to the best of my knowledge and belief:

ADDITIONAL
COMMENTS:_____

_____ _____

System Manager Date

I certify that the information provided above is true and correct to the best of my knowledge and belief:

ADDITIONAL
COMMENTS:_____

_____ _____

System Customer Date

APPENDIX C: March 12, 1997 Memorandum: "Year 2000 Refined Reporting Requirements for DoD and Quarterly Reporting Checklists

Appendix D Acronyms

AIS - Automated Information System

ANSI - American National Standards Institute

ASCII - American Standard Code for Information Interchange

ASD - Assistant Secretary of Defense

BIOS - Basic input/output system

BPR - Business Process Reengineering

C3I - Command, Control, Communications, and Intelligence

CC - Century indicator

CIO - Chief Information Officer

CFS - Center for Standards

COTS - Commercial off-the-Shelf

COE - Common Operating Environment

CPU - Central Processing Unit

DASD Deputy Assistant Secretary of Defense

DBMS - database management systems

DDRS - Defense Data Dictionary System

DII - Defense Information Infrastructure

DISA - Defense Information Systems Agency

DIST - Defense Integration Support Tools

DoD - Department of Defense

DODD - DoD Directive

DOS - Disk Operating System

EC - Electronic Commerce

EDI - Electronic Data Interchange

ELOC - Executable Line of Code

FIPS - Federal Information Processing Standard

GAO - General Accounting Office

GOTS - Government off-the-Shelf

GPS - Global Positioning System

IAW - In accordance With

IEEE - Institute for Electrical and Electronics Engineers

ISO - International Organization for Standardization

LAN - Local Area Network

LANIWAN - Local Area Network in a Wide Area Network

LOC - Lines of Code

MIL-STD - Military Standard

MOA - Memorandums of Agreement

NIST - National Institute of Standards and Technology

NSA - National Security Agency

O/S - Operating System

OSD - Office of the Secretary of Defense

OSE - Open System Environment

PEO - Program Execution Officer

PM - Program Manager

POC - Point of Contact

PUB - Publication

RAAD - Rapid Architecture Application Development

RAD - Rapid Application Development

RTC - Remote Terminal Controllers

STD - Standard

TAFIM - Technical Architecture Framework for Information Management

USD - Under Secretary of Defense

USMTF - Uniform Services Message Text Format

WAN - Wide Area Network

WWW World Wide Web

Y2K - Year 2000

Appendix E Glossary

calendar errors: errors typically include failing to treat 2000 as a leap year and converting incorrectly between date representations.

certified system: For purposes of this plan only, certified system is a system which the system administrator has signed off on as compliant via the checklist in Appendix B.

compliant: "compliant" systemÆs dates are stored, manipulated (including, but not limited to calculating, comparing, and sequencing), exchanged, and displayed in a way that cannot be misinterpreted and are not ambiguous. "Compliant" systemÆs hardware and software productsÆ correctly process date and date related data individually and in combination in both the 20th and 21st Centuries. Finally, "compliant" systems have no extended semantics, calendar errors, date overflow, and inconsistent semantics.

contingency plan: a plan for responding to the loss of system use due to a disaster such as a flood, fire, computer virus, or major software failure. The plan contains procedures for emergency response, backup, and post-disaster recovery.

conversion: the process of making changes to databases or source code.

database: an aggregation of data; a file consisting of a number of records or tables, each of which is constructed of files or a particular type, together with a collection of operations that facilitate searching, sorting, recombination, and similar operations.

data overflow: many software products represent dates internally as a base date/time plus an offset in days, seconds, or microseconds since that base date/time. Hardware integers holding the offset value can overflow past the maximum corresponding dateùan event that may lead to undefined behaviors.

Defense Integration Support Tools (DIST): A tool set developed by Defense Information Systems Agency (DISA) to support the DoD-wide information management requirement and provide a migration planning and assessment decision support capability.

executable lines of code (ELOC): source lines of code minus comments, white-space and data declarations. The unit of measure used in costing models to capture the effort to create the functional portion of a software program. Used to cost out the effort to develop the functionality.

extended semantics: in general, specific values for a date field are reserved for special interpretation.

The most common example is interpreting "99" in a 2-digit year field as an indefinite end date, i.e., "does not expire." Another is embedding a date value in a *non-date* data element.

inconsistent semantics: at interface between systems, software on each side assumes semantics of data passed. Software must make same century assumptions about 2-digit years.

interface: a boundary across which two systems communicate. An interface might be a hardware connector used to link to other devices, or it might be a convention used to allow communication between two software systems. This is to include interfaces internal to the system, its applications and programs, to other internal or external systems?

integration: two or more software applications that must run on the same physical processor(s) and under the same operating system.

integration testing: testing to determine that the related information system components perform to specification.

interoperability: (1) The ability of two or more systems or components to exchange data and use information (IEEE STD 610.12) (2). The ability of two or more systems to exchange information and to mutually use the information that has been exchanged.

legacy system: An existing automated information system (AIS) or application which will be replaced in whole or part by a migration system.

line of code: a single computer program command, declaration, or instruction. Program size is often measured in lines of code.

migration system: An existing automated information system (AIS) or application, or a planned and approved AIS or application, that has been officially designated as the single AIS or application to support standard processes for a function.

mission critical system: a system that when its capabilities are degraded, the organization realizes a resulting loss of a core capability.

object code: the machine code generated by a source code language processor such as an assembler or compiler. A file of object code may be immediately executable or it may require linking with other object code files, e.g. libraries, to produce a complete executable program.

parallel processing: the simultaneous use of more than one computer to solve a problem.

regression testing: selective retesting to detect faults introduced during modification of a system.

risk assessment: a continuous process performed during all phases of system development to provide an estimate of the damage, loss, or harm that could result from a failure to successfully develop individual system components.

risk management: a management approach designed to reduce risks inherent to system development.

system testing: testing to determine that the results generated by the enterpriseÆs information systems and their components are accurate and the systems perform to specification.

test facility: a computer system isolated from the production environment dedicated to the testing and validation of applications and systems components.

unit testing: testing to determine that individual program modules perform to specification.

validation: the process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements.

Year 2000 compliant: information systems able to accurately process date data--including, but not limited to, calculating, comparing, and sequencing--from, into, and between the twentieth and twenty-first centuries, including leap year calculations.

Year 2000 problem: the potential problems and its variations that might be encountered in any level of computer hardware and software from microcode to application programs, files, and databases that need to correctly interpret year-date data represented in 2-digit-year format.

Year 2000 (Y2K) system: An automated process that uses information technology such as computer hardware and software to perform a specific function, application, or service.

Appendix F References

Secretary of Defense "All Hands" Year 2000 Message, Year 2000 System Failures began the formal awareness phase of DoD five phase management process, dated November 27, 1995.

ASD(C3I) Memorandum, Subject: Y2K Computing Problem with Personal Computers (PC) and Workstations, dated May 8, 1996.

ASD(C3I) Memorandum, Subject: Y2K Assessment, dated Aug. 1, 1996.

48 Code of Federal Regulations, Parts 39.002 et. seq.

Deputy Secretary of Defense Memorandum, Subject: Y2K Date Processing, dated Aug. 16, 1996.

US Army Project Change of Century Action Plan, dated Oct. 4, 1996.

ASD(C3I) Memorandum, subject: System Interfaces, Data Exchanges, and Defense Integration Support Tools requires systems and their interfaces to be registered in the DIST database, dated November 5, 1996.

GAO: Y2K Computing Crisis: An Assessment Guide, Exposure Draft, February 1997, available at GAO.

ASD(C3I) Memorandum, Subject: Year 2000 (Y2K) Refined Reporting Requirements for DoD, dated March 12, 1997.

Federal Information Processing Standards (FIPS)4-1: "Representation for Calendar Date and Ordinal Date for Information Interchange," March 25, 1996.

Federal Year 2000 Web Sites

Year 2000 Interagency Committee

http://www.itpolicy.gsa.gov/mks/yr2000/y201toc1.htm

Army

http://www.army.mil/army-y2k/

Air Force

http://infosphere.safb.af.mil/~jwid/fadl/world/y2k.htm

Navy

http://www.nismc.navy.mil/horizon/year2000/year2000.htm

Marine Corps

http://issb-www1.mqg.usmc.mil/year2000/

Defense Information Systems Agency

http://www.disa.mil/cio/y2k/cioosd.html

Defense Information Systems Agency Tools

http://www.mitre.org/research/y2k/docs/TOOLS_CAT.html

Air Force Software Technology Support Center

http://www.stsc.hill.af.mil/RENG/idex.html

Army Tools

http://www.army.mil/army-y2k/tools/tools~1.htm

MITRE Homepage including a list of Y2K software and hardware products which have NOT been validated for compliance.

http://www.mitre.org/research/cots/COMPLIANCE_CAT.html

The root address for legal issues:

http://www.year2000.com "legal issues concerning the year 2000 millennium bug."

http://www.year2000.com/archive/beyond.html

http://www.year2000.com/archive/legal.html